

Automatic Crowd Density and Motion Analysis in Airborne Image Sequences Based on a Probabilistic Framework

Anonymous ICCV submission

Paper ID ****

Abstract

Real-time monitoring of crowded regions has crucial importance to avoid overload of people in certain areas. Understanding behavioral dynamics of large people groups can also help to estimate future status of underground passages, public areas, or streets.

In order to bring an automated solution to the problem, we propose a novel approach using airborne image sequences. Our approach depends on extraction of local features from invariant color components of the images. Using extracted local features as observations, we form probability density functions (pdf) for each image of input sequence which holds information about density of people. We introduce four measures to extract information about pdf characteristics. A change within the four measures over the image sequence gives important information about status of the crowds. Besides, we also use obtained pdfs to estimate main crowd motion directions. To test our algorithm, we use a stadium entrance image data set, and two festival area data sets taken from an airborne camera system. In order to be later able to reach real-time performance the algorithms use parameters which can be extracted directly from the image data. Experimental results indicate possible usage of the developed algorithms in real-life events.

1. Introduction

Recently automatic detection of dense crowds and understanding people behaviors from images has become a very important research field, since it can provide crucial information especially for police departments and crisis management teams.

In order to monitor large outdoor events, researchers tried to develop algorithms which can work on outdoor camera images or video streams. Arandjelovic [1] developed a local interest point extraction based crowd detection method to classify terrestrial images as crowd and non-crowd regions. They observed that dense crowds produce a high

number of interest points. Therefore, they used density of SIFT features for classification. After generating crowd and non-crowd training sets, they used SVM based classification to detect crowds. They obtained scale invariant results on terrestrial images. Unfortunately, these images are not useful to monitor large events, and the classifier has to be trained before hand. Chao et al. [8] tried to obtain quantitative measures about crowds using single images. They used Haar wavelet transform to detect head-like contours, then using SVM they classified detected contours as head or non-head regions. They provided quantitative measures about number of people in crowds and sizes of crowds. Although results are promising, this method requires clear detection of human head contours and a training of the classifier. Understanding motion of people, and crowds using image sequences has very high importance to prevent undesirable events. To this end, Doulamis proposed a particle filtering based robust approach for tracking multiple people even under partial or full occlusion scenarios. Promising results are obtained using indoor image sequences [6]. Bremond and Thonnat [2], developed a multiple people tracking system using outdoor cameras. Chellapa et al. [13] used multiple outdoor cameras to generate a 3D scene model. Using background subtraction from each image in sequence, they obtained changing pixels which are used as an input to their multiple object tracking system. Unfortunately, street cameras have a limited coverage area to monitor large outdoor events. In addition to that, in most of the cases, it is generally not possible to obtain close-range street images or video streams in the place where an event occurs. Therefore, to monitor crowded regions in very big outdoor events, the best way is to use airborne images. Since most of the previous approaches in this field needed clear detection of face or body features, curves, or boundaries to identify crowds which is not possible in airborne images, new approaches are needed to extract information from these images. In a previous study Hinz et al. [7] registered airborne image sequences to estimate density and motion of people in crowded regions. For this purpose, first a training background segment was selected manually to classify image

as foreground and background pixels. They used the ratio of background pixels and foreground pixels in a neighborhood to plot a density map. Observing change of the density map in the sequence, they estimated motion of people. However, their approach did not provide quantitative measures about crowds. As it is discussed in their article, understanding crowd motion has also crucial importance. In a following study, Burkert et al. [3] used their previous approach to detect individuals. Positions of detected people are linked with graphs. They used these graphs for understanding the behavior of people. Courty and Corpetti [5], used optical flow to estimate motion patterns in the image sequences. Using this information, they generated 3D crowd simulations. Maury and Venel [9], introduced mathematical models to model human behavior especially for emergency cases.

Herein, we propose a fully automatic framework to detect and the analyze behavior of crowds using airborne image sequences. In Fig. 1.(a), we give an example airborne image which is taken over an outdoor festival site. The image is taken from a Cesna aircraft from approximately 1000 m. flight altitude, exhibiting a field of view of about 800 m with a Canon EOS Mark II camera. In Fig. 1.(b), we represent a subpart of the image which is zoomed into a complex region with people and tents. In In Fig. 1.(c), we represent a closer view of a dense crowd region in front of a concert stage. As can be seen in this example, it is not possible to detect head or body features due to the spatial resolution of the airborne image. However, we can still notice a change of color components in the place where a person is present. Therefore, herein we propose a color feature based framework to monitor crowded regions. Using features as observation points, we form a probability density function (pdf) for each image of the input sequence. Analyzing image pdfs, we extract information about status and also motion of the crowds. Obtained experimental results with self-adapting system parameters show reliability of the proposed system.

2. Local Feature Extraction

For local feature extraction, we use features from accelerated segment test (FAST). FAST feature extraction method is especially developed for corner detection purposes by Rosten et al. [12], however it also gives responses on small regions which are significantly different than surrounding pixels. Sirmacek and Unsalan experimented usage of this feature extraction method for detecting object characteristics in satellite images [14]. Their test results proves that FAST features can be used to extract important interest locations in remotely sensed images.

We use invariant color bands of the image for FAST feature extraction. We first start with converting our RGB test image into CIELab color space. CIELab color space bands

are able to enhance different colors best and minimize color variances. After transforming the RGB color image into CIELab color space, again we obtain three bands as L , a , and b [11]. Here, L band corresponds to the intensity of the image pixels. a , b bands contain chroma features of the image. For illumination invariance, we use only a and b chroma bands of the image for local feature extraction. We assume (x_a, y_a) $a \in [1, 2, \dots, K_a]$ and (x_b, y_b) $b \in [1, 2, \dots, K_b]$ as FAST local features which are extracted from a and b chroma bands of the image respectively. Here, K_a and K_b indicate the maximum number of features extracted from each chroma band. As local feature, in our study we use (x_i, y_i) $i \in [1, 2, \dots, K_i]$ which hold features coming from two chroma bands. However, if two features from different bands are extracted at the same coordinates, it is held only for one time in (x_i, y_i) $i \in [1, 2, \dots, K_i]$ array. We represent locations of detected local features for a sub-region of *Festival₂* test image in Fig. 1.(e). Extracted FAST features will behave as observations of the probability density function (pdf) of dense crowds to be estimated, after a feature selection process which we introduce in the next step.

3. Local Feature Selection

As can be seen in Fig. 1.(e), we detected FAST features at almost each individual person position. Unfortunately, corners of other objects also lead to FAST feature detection. We apply segmentation to the input image in order to estimate the interest region which helps us to eliminate redundant local features. For segmentation, we benefit from the mean shift segmentation approach which was proposed by Comanicu and Meer [4]. At mean shift segmentation process, we choose spatial bandwidth (h_s) and spectral bandwidth (h_r) parameters as 7 and 6.5 respectively after extensive tests, and we use the same parameters for all input images. The segmentation result is a new image called $S(x, y)$ which holds each segment labeled by a single color. We present the mean shift segmentation result for our *Festival₂* test image in Fig. 1.(d). Here, each segment is labeled with the mean of red, green, and blue band values of the original image pixels which are inside of the segment. Although we have no idea about which segment represents which object, the segmentation result can be useful to decrease the complexity of the problem. We believe that, on the interest region (generally roads or paved areas) there should be many local features indicating people. Therefore, we eliminate regions having less than 50 local features inside. Remaining regions are assumed as interest regions which is represented with value 1 in $M(x, y)$ binary mask. In the next steps of the algorithm, we use (x_i, y_i) $i \in [1, 2, \dots, K_i]$ local features only if they satisfy $M(x_i, y_i) = 1$ equation. In Fig. 1.(f), we represent feature selection result on a sub-region of *Festival₂* test image. As

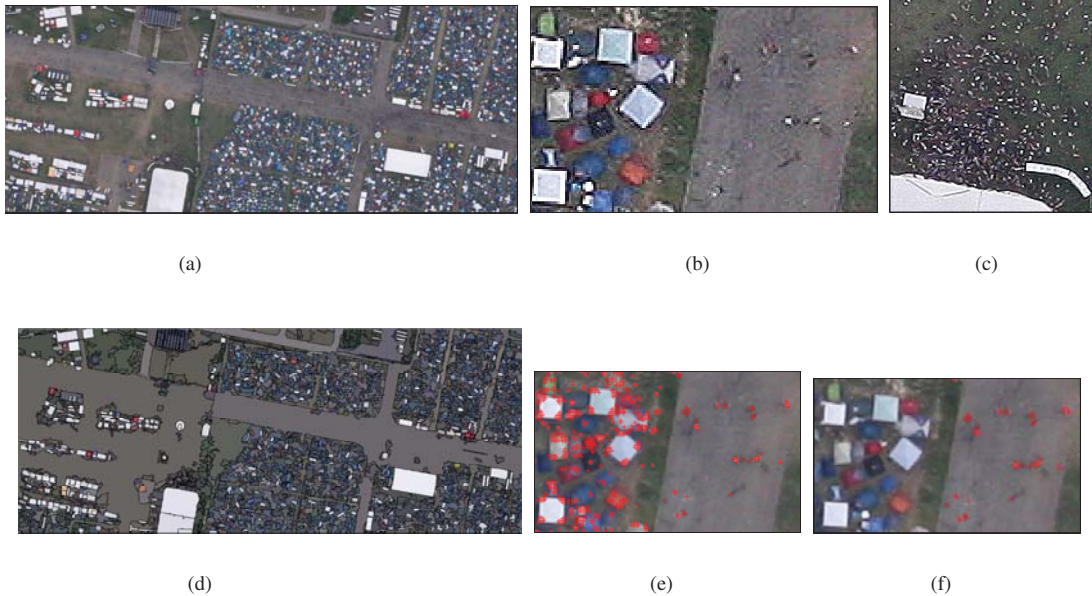


Figure 1. (a) A test image ($Festival_2$) from our festival airborne image sequence, (b) Closer view of a region in $Festival_2$ image in order to represent real image resolutions, (c) Closer view of a dense crowd region in $Festival_2$ image, (d) Mean shift segmentation result on $Festival_2$ test image, (e) Extracted (x_i, y_i) local features on $Stadium_1$ test image (closer view of a sub-region is represented), (f) Remaining local features after feature selection test for the same sub-region.

can be seen in this result, redundant local features coming from other structures are eliminated successfully. Next, we introduce an adaptive kernel density estimation method, to estimate the corresponding pdf which will help us to detect dense people groups and people in sparse groups.

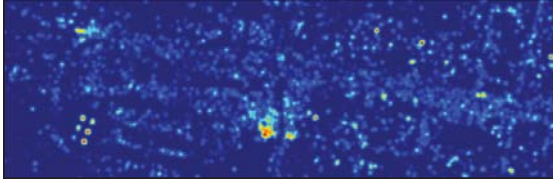
4. Adaptive Kernel Density Estimation for Crowd Detection

Since we have no pre-information about crowd locations in the image, we formulate the crowd detection method using a probabilistic framework. We assume each FAST feature as an observation of a crowd pdf. For crowded regions, we assume that more local features should come together. Therefore, knowing the pdf will lead to detection of crowds and will also give information about the density of people. For pdf estimation, we benefit from a kernel based density estimation method. Using Gaussian symmetric kernel functions, the estimated pdf is formed as below;

$$p(x, y) = \frac{1}{R} \sum_{i=1}^{K_i} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma^2}\right) \quad (1)$$

where σ is the bandwidth (smoothing parameter) of Gaussian kernel, and R is the normalizing constant to normalize $p(x, y)$ values between $[0, 1]$. In kernel based density

estimation the main problem is how to choose the bandwidth of Gaussian kernel, since the estimated pdf directly depends on this value. For instance, if the resolution of the camera increases or if the altitude of the plane decreases, pixel distance between two persons will increase. That means, Gaussian kernels with larger bandwidths will make these two persons connected and will lead to detect them as a group. Therefore, the bandwidth of Gaussian kernel should be adapted for any given input image. In probability theory, there are several methods to estimate the bandwidth of kernel functions for given observations such as statistical classification based methods, and balloon estimators. Unfortunately, those approaches require very high computation time especially when many observation point exists. For time efficiency, we follow an approach which is slightly different from balloon estimators. First, we pick $K_i/2$ number of random observations to reduce the computation time. For each observation location, we compute the distance to the nearest neighbor observation point. Then, the mean of all distances give us a number l . We assume that the variance of Gaussian kernel (σ^2) should be equal or greater than l . In order to guarantee the intersection of kernels of two close observations, we assume the variance of Gaussian kernel as $5l$ in our study. For a given sequence, that value is computed only one time over one image. Then, the same σ value is used for all images of the same sequence.



(a)



(b)

Figure 2. (a) Probability density function for *Festival₂* test image. (b) Dense crowd detection result on *Festival₂* test image (dense people groups in front of the concert stages are detected).

Our automatic kernel bandwidth estimation method, makes the algorithm robust to scale and resolution changes. After calculating $p(x, y)$, we use Otsu's automatic thresholding method on this pdf to detect regions having high probability values which indicates crowded regions and is stored in a binary image ($B(x, y)$) [10]. After thresholding, regarding resolutions of the input data, binary regions smaller than 1000 pixels are eliminated since they cannot indicate large crowds. In Fig. 2.(a) and Fig. 2.(b), we represent pdf function and detected dense crowd boundaries respectively for *Festival₂* image.

5. Crowd Density Analysis

After we obtain pdfs $p_n(x, y)$ $n \in [1, 2, \dots, N]$, where n indicates the image index in a sequence, we introduce four measures on them to analyze crowd density.

5.1. Using Number of People in Crowd Region as Density Measure

We want to extract quantitative measures about approximate number of people from detected crowd regions for more detailed analysis. Unfortunately, number of local features in a crowd region do not give the number of people directly. In most cases, shadows of people or small gaps between people also generate a feature. Besides, two neighbor features might come from two different chroma bands of the same person. In order to decrease counting errors, we follow a different strategy to estimate the number of people. We use a binary mask $F_n(x, y)$ for n th im-

age in the sequence where feature locations have value 1. Then, we dilate $F_n(x, y)$ binary image using a disk shape structuring element with a radius of 2 to connect close feature locations. Slight change of radius of the structuring element does not make a significant change in estimated people number. However, an appreciable increase in radius can connect features coming from different persons, and that leads to underestimations. Finally, we apply connected component analysis to $F_n(x, y)$ mask, and count the number of connected components appearing in each dense crowd region represented in $B_n(x, y)$. For j th crowd region in $B_n(x, y)$, the number of connected components inside are stored as the first measure in $M_1^j(n)$ array.

5.2. Using Size of Crowd Region as Density Measure

As the number of people increases, the size of the extracted crowd region also increases. Therefore, the size of j th crowd region in n th image of the sequence which is stored in $B_n^j(x, y)$ can be used as a measure for crowd analysis. Besides, this measure is not directly effected by over or under estimations of the number of people. Our second measure can be calculated as follows;

$$M_2^j(n) = \sum_{x=1}^X \sum_{y=1}^Y B_n^j(x, y) \quad (2)$$

Here, X and Y represents the size of the input image in x and y direction.

5.3. Using Sum of Probabilities in Crowd Region as Density Measure

As third measure, we count the sum of non-normalized pdf values for each crowd region. We assume that the sum of probability values in a crowd region gives an idea about both density and number of people. For j th crowd region in n th image of the sequence, it is calculated as follows;

$$M_3^j(n) = \sum_{x=1}^X \sum_{y=1}^Y B_n^j(x, y) \times R \times p_n(x, y) \quad (3)$$

5.4. Using Fused Measures

Finally, we fuse the proposed density measures. For fusing extracted $M_1^j(n)$, $M_2^j(n)$, and $M_3^j(n)$ for j th dense crowd region, each input measure is normalized between $[0, 1]$, and their mean is computed. Here, for j th crowd region we represent fused measures with $M_4^j(n)$ array which exhibits values in $[0, 1]$ interval. As the previous three measures, increase of $M_4^j(n)$ values indicate increase of unpleasant crowd densities.

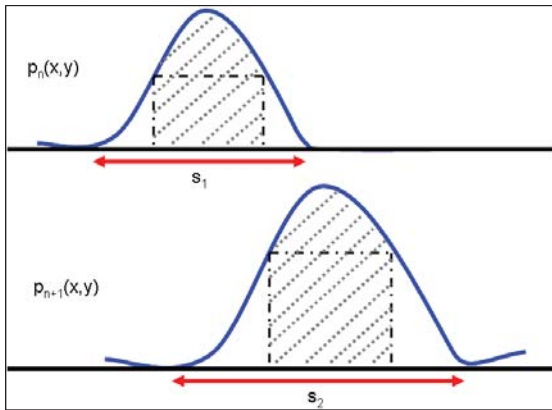


Figure 3. Demonstration of crowd motion analysis using pdf functions of images in the sequence. (One-dimensional slices of pdfs are assumed to simplify representation.)

6. Crowd Motion Analysis

In this step, we introduce a model for crowd motion analysis using pdfs of images in the sequence. If a crowd is moving, the local maximum in pdf will be shifted towards the main movement direction. In Fig. 3, we illustrate the crowd motion analysis idea. Here, we assume that the first function represents a slice of two-dimensional pdf of n th image in the sequence, and the second function represents a slice of two-dimensional pdf of $(n+1)$ th image. We assume that crowd movement direction is from the local maximum of the first function, to the closest local maximum of the second function.

According to our experimental observations, here we can also add one additional note. Besides using our four measures, we believe that two more properties from these pdf functions can give information about the status of crowds. The area under crowd pdf, which is shown with s_1 and s_2 measures in one-dimension, can give an idea about the dense crowd region size. As second property, a discrete integral of the shadowed region, which shows the area between two sides of the pdf function where the function value drops to 0.6 of the local maximum value can be calculated. We believe that these discrete integral values (assumed as a_1 and a_2 respectively), can give further information about crowd density. However, in this study we focus on only usage of the proposed M_{1-4} measures.

7. Experiments

7.1. Crowd Density Detection Performance

In order to evaluate our experimental results, first we discuss our dense crowd detection and quantitative estimation performances. In Fig. 6, we provide detection results for the stadium entrance data set which includes 14 images. In

Fig. 6.(a), (b), and (c), we provide detected dense crowd boundaries for *Stadium*₁, *Stadium*₈, and *Stadium*₁₄ images respectively. Our sequence includes four dense crowds in front of the entrances. In order to prove robustness of our quantitative measurements, in this part we compare our automatic measurements and our reference data. Measurements are obtained over four dense crowds in front of four entrance gates from left to right respectively. Since even for a human observer it is hard to count the exact number of people in crowds, we have assumed mean of counts of three human observers as groundtruth. In Table 1, we compare automatically detected number of people (N), and density (d) with reference data (N_{ref} and d_{ref} respectively) for each crowd. Similarity of our measures with reference data which is obtained by calculating the mean of three human observer responses. Here, red square labeled bars indicate reference values (N_{ref}), and blue circle labeled bars indicate automatically obtained number of people in dense crowd regions (N).

Table 1. Comparison of reference data and automatically detected people number and density estimation results for test regions in *Stadium*₁.

	REGION ₁	REGION ₂	REGION ₃	REGION ₄
N	139	211	115	102
N_{ref}	132	180	114	98
d	0.81	0.74	0.68	0.76
d_{ref}	0.76	0.63	0.67	0.73

7.2. Evaluation of Obtained Measures and Main Crowd Motion Directions

We test our four novel measures on three different image sequences. In two test image sequences, congestion at the entrances increases over time. In Fig. 5.(a), we represent our Oktoberfest image sequence in the first row. In the second row, we represent corresponding $p_{1-4}(x,y)$ pdf's. In the last row, we represent detected very dense crowd boundaries. In Fig. 5.(b), we represent M_4^1 fused measure values for detected crowd region. A very rapid increase of M_4^1 measure for the last image indicates a sudden density increase of people. In Fig. 5.(c), we represent automatically estimated main motion directions.

For our stadium entrance image sequence, in Fig. 6.(d), (e), (f), and (g), we provide M_4^1 measures obtained from the whole image sequence for each dense crowd region separately. Results indicate an increase of density and size of crowds over time. In this sequence, however individuals are moving, the dense crowd positions do not change. Therefore, we do not talk about main motion directions on this data set.

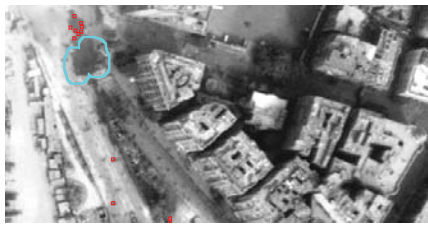


Figure 4. Dense crowd detection result on a WorldView2 satellite image is represented to prove robustness of system parameters to the spatial resolutions of the input data.

Finally, we focus on a sub-region for the festival area that we represented in Fig. 1.(a). In the first image of the sequence, people are walking towards the concert stage. During concert (at the second and the third image), the main motion direction does not change. Finally, at the last image, the crowd moves away from the concert area. In Fig. 7.(e), we represent extracted M_4 measures. Sudden increase of fused measures for the second image indicate the sudden rise of unpleasant crowd density. Again a sudden increase in fused measures can be seen for the dispersing crowd after the concert event. In Fig. 7.(d) we represent estimated main motion directions for the dense crowd in this sequence. Automatically detected main crowd motion directions show the motion towards the concert stage, and back to the festival area after event.

7.3. Self Adaptation of the System Parameters

The main advantage of the proposed system is it is capability to adapt itself to spatial resolutions of the input images automatically which is important if the algorithms will be used in a real-time environment. Since all crowd monitoring measures and main crowd motion estimations depend on the generated pdf, it is very important to obtain a pdf which represents dense crowd groups independently from input image resolutions. Automatic estimation of the best σ value is provided as an input value to Eqn. 1. In order to prove the robustness of the system to the spatial resolutions of the input images, we present a crowd detection example on a WorldView-2 satellite image in Fig. 4. For processing the satellite image, only a panchromatic channel is used. In this image, the blue boundary represents detected dense crowd region. We use red squares to represent positions of other local features which are laying outside of the dense crowd boundaries since they can indicate individual people or very small people groups. Obtained result on satellite image which have approximately 0.5 m. spatial resolution indicates the robustness of automatically adapted system parameters.

7.4. Computation Time

Computation time requirements of the proposed system is also reasonable. Using an Intel Core2Quad 2.66GHz PC and MatLab programming environment for implementation, we extracted four measures for each Oktoberfest input image (represented in the first row of Fig. 5.(a)) in 8.05 seconds per image as an average value. In real-time applications, we plan to achieve higher computation speeds by implementing the system in C programming environment.

8. Conclusions

In order to solve crowd monitoring problem, herein we proposed a novel approach to detect dense crowds and to watch their status automatically using airborne image sequences. To this end, we developed a local invariant feature extraction based probabilistic approach. To watch the status of the detected dense crowds over time, we proposed four novel measures which are extracted from automatically generated probability density functions of images in the sequence. Changes of extracted measures are observed over the sequence to understand densities and sizes of crowds. Using probability density functions, we also estimated main motion directions of crowds. We tested our algorithm on three image sequences which are taken from three different events.

References

- [1] O. Arandjelovic. Crowd detection from still images. *British Machine Vision Conference (BMVC'08)*, Sep. 2008. 1
- [2] F. Bremond and M. Thonnat. Tracking multiple nonrigid objects in video sequences. *IEEE Transactions on Circuits and Systems for Video Technology*, 8 (5):585–591, 1998. 1
- [3] F. Burkert, F. Schmidt, M. Butenuth, and S. Hinz. People tracking and trajectory interpretation in aerial image sequences. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (IAPRS), Commission III (Part A)*, XXXVIII:209–214, Sep. 2010. 2
- [4] D. Comanicu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002. 2
- [5] N. Courty and T. Corpetti. Crowd motion capture. *Computer Animation and Virtual Worlds*, 18:361–370, 2007. 2
- [6] A. Doulamis. Dynamic tracking re-adjustment: a method for automatic tracking recovery in complex visual environments. *Multimedia Tools and Applications*, 50 (1):49–73, 2010. 1
- [7] S. Hinz. Density and motion estimation of people in crowded environments based on aerial image sequences. *ISPRS Hannover Workshop on High-Resolution Earth Imaging for Geospatial Information*, 2009. 1
- [8] S. Lin, J. Chen, and H. Chao. Estimation of number of people in crowded scenes using perspective transformation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 31 (6):645–654, Nov. 2001. 1

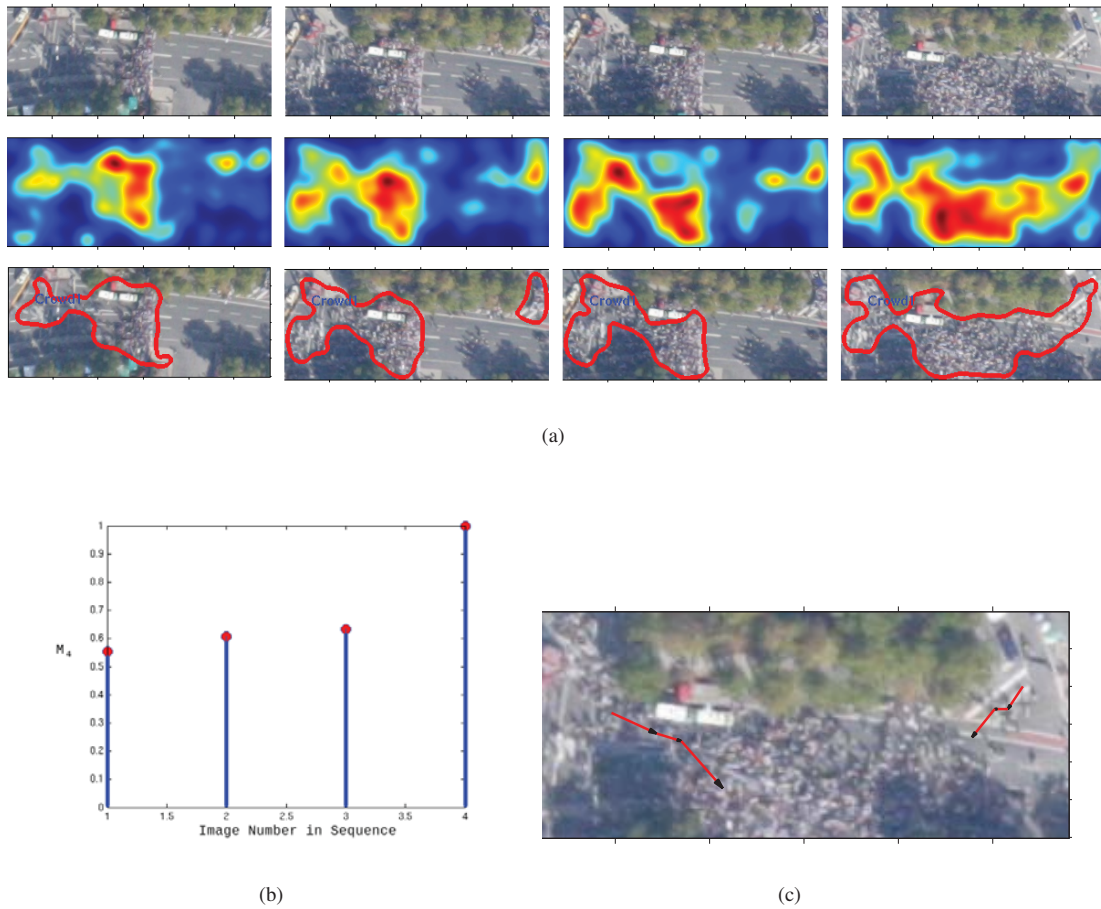


Figure 5. (a) $Oktoberfest_{1-4}$ images from Oktoberfest image sequence is represented in the first row, corresponding $p_{1-4}(x, y)$ pdf's are represented at the second row, detected dense crowd boundaries are represented in the last row. (b) Calculated M_4 measure is plotted for each image in the sequence. (c) Estimated main motion directions are plotted on $Oktoberfest_4$ image.

- [9] B. Maury and J. Venel. Handling of contacts in crowd motion simulations. *Traffic and Granular Flow'07, Springer, Berlin, Heidelberg*, 1:171–180, 2007. 2
- [10] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on System, Man, and Cybernetics*, 9 (1):62–66, 2009. 4
- [11] G. Paschos. Perceptually uniform color spaces for color texture analysis: an empirical evaluation. *IEEE Transactions on Image Processing*, 10:932–937, 2001. 2
- [12] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Learning*, 32 (1):105–119, Nov. 2010. 2
- [13] A. Sankaranarayanan, R. Patro, P. Turaga, A. Varshney, and R. Chellappa. Modeling and visualization of human activities for multicamera networks. *EURASIP Journal on Image and Video Processing*, 2009. 1
- [14] B. Sirmacek and C. Unsalan. Using local features to measure land development in urban regions. *Pattern Recognition*

Letters, 31 (10):1155–1159, July 2010. 2

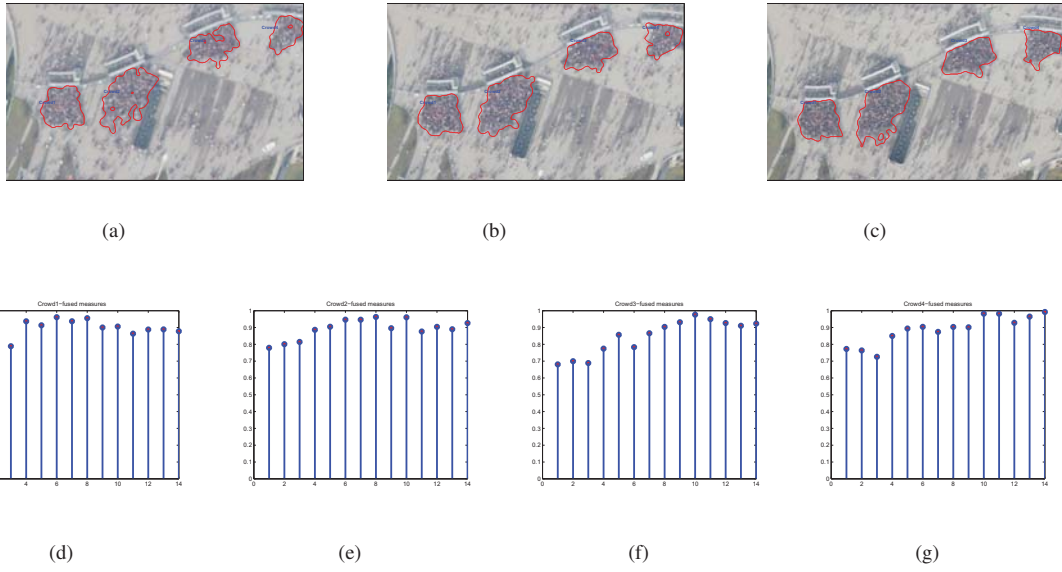


Figure 6. (a) Detected crowds for *Stadium*₁, (b) Detected crowds for *Stadium*₈, (c) Detected crowds for *Stadium*₁₄, (d) M_4^1 measure values, (e) M_4^2 measure values, (f) M_4^3 measure values, (g) M_4^4 measure values.

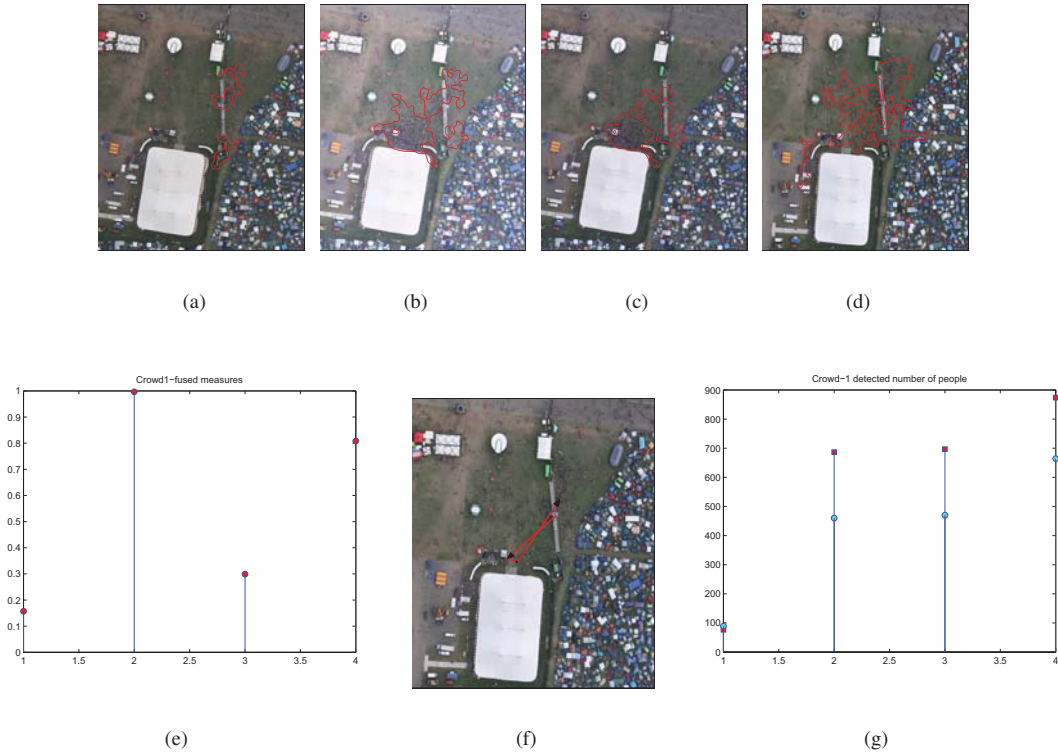


Figure 7. (a) *Concert*₁ image and detected dense crowd, (b) *Concert*₂ image and detected dense crowd, (c) *Concert*₃ image and detected dense crowd, (d) *Concert*₄ image and detected dense crowd, (e) M_4 measures, (f) Estimated main crowd motion directions, (g) Comparison of estimated number of people in crowds with reference data.